

Advanced Financial Instruments for Sustainable Business and Decentralized Markets

Prof. Abhinava Tripathi

Department of Management Sciences

Indian Institute of Technology Kanpur

Lecture 31

Week 11

In this lesson, we will discuss the key properties of digital signatures including asymmetry and key cryptography. We will also understand consensus mechanism that securitizes the blockchain transactions. Some of the important topics as part of consensus mechanism include Byzantine Generals Problem, Proof of Work and Proof of Stake mechanism, Incentive Structures in Proof of Work and Proof of Stake mechanisms. In this video, we will discuss digital signatures and asymmetry, and key cryptography. Regardless of how the data is formed and transacted, determining the validity and authenticity of a transaction is very important. The validity of a transaction shows that the transaction meets the protocol requirements and any formalized data formats or for example smart contract specific requirements to the blockchain implementation.



Digital Signature

- Regardless of how the data is formed and transacted, determining the validity and authenticity of a transaction is important.
- For digital signatures, two properties that correspond well to the handwritten signature analogy:
 1. Only you can sign: only you can make your signature, but anyone who sees it can verify that it's valid.
 2. Tied to a document: so that the signature cannot be used to indicate your agreement or endorsement of a different document

The authenticity of a transaction is also important as it determines that the sender of digital assets had access to those digital assets. This requires digital signatures to determine that the sender of digital assets had access to those digital assets. This requires digital signatures which is another very important cryptographic primitive along with the hash functions that we need as building blocks for the cryptocurrency discussion. A digital signature is supposed to be the digital analogue of handwritten signature on paper and transactions are typically digitally signed by the sender's associated private key and can be verified at any time using the associated public key.

Now we desire two key properties from digital signatures that correspond well to the handwritten signature analogy. First, it is that only you can sign or make your signature but anyone who sees it can verify that it is valid. Second, they should be tied to a document. We want the signature to be tied to a particular document so that the signature cannot be used to indicate your agreement or endorsement of a different document. For handwritten signatures, this latter property is analogous to assuring that somebody can't take your signature and snipe it off one document and paste it onto the bottom of another document.



Asymmetric Key Cryptography

- Blockchain technology uses asymmetric-key cryptography : a public key and a private key that are mathematically related to each other
- Asymmetric-key cryptography enables a trust relationship between users who do not know or trust one another
- This contrasts with symmetric-key cryptography in which a single secret key is used to both encrypt and decrypt

The previous discussion strongly hinges on the subject called asymmetric key cryptography. Now blockchain technology uses asymmetric key cryptography which is also referred to as public key cryptography often. Asymmetric key cryptography uses a pair of keys notably a public key and a private key that are mathematically related to each other. The public key here is made public without reducing the security of the process but the private key must remain secret if the data is to retain its cryptographic protection. Even though there is a relationship between the two keys, the private key cannot efficiently be determined based on the knowledge of the public key.



Digital Signature Scheme

- Digital signature scheme:
- ❖ $(sk, pk) := generateKeys(keysize)$.
- The secret key sk is kept privately and used to sign messages. pk is the public verification key that you give to everybody.
- ❖ $sig := sign(sk, message)$
- The sign method takes a message and a secret key, sk , as input and outputs a signature for the message under sk .
- ❖ $isValid := verify(pk, message, sig)$
- The verify method takes a message, a signature, and a public key as input. It will be true if sig is a valid signature for the message under public key pk and false otherwise.

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

6

And one can encrypt with the private key and then decrypt with the public key. Alternately, one can encrypt with the public key and then decrypt with the private key. Now this asymmetric key cryptography enables a trust relationship between users who do not know or trust each other. By providing a mechanism to verify the integrity and authenticity of transactions while at the same time allowing transactions to remain public. And to do this, the transactions are digitally signed.



Digital Signature: Security Properties

- Two important security properties:
- Valid signatures must verify: $verify(pk, message, sign(sk, message)) == true$
- Infeasible to find Private Key (SK) from Public Key (PK)
- Signatures are unforgeable: The second requirement is that it's computationally infeasible to forge signatures. That is, an adversary who knows your public key and gets to see your signatures on some other messages can't forge your signature on some message for which he has not seen your signature.

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

7

This means that a private key is used to encrypt a transaction such that anyone with the public key can decrypt it. Since the public key is freely available, encrypting the transaction

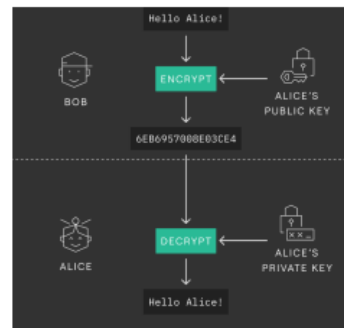
with the private key proves that the signer of the transaction has access to the private key as well. And alternatively, one can encrypt data with the user's public key such that only users with access to private key can decrypt it. A drawback here is that this asymmetric key cryptography is often slow to compute. Now this contrasts with the symmetric key cryptography in which a single secret key is used to both encrypt and decrypt.

With symmetric key cryptography, users must already have a trust relationship established with one another to exchange the pre-shared key. In a symmetric system, any encrypted data that can be decrypted with the pre-shared key confirms that it was sent by another user with access to the pre-shared key. No user without access to the pre-shared key will be able to view the decrypted data compared to asymmetric key cryptography. Symmetric key cryptography is very fast to compute. Because of this, when one claims to be encrypting something using asymmetric key cryptography, often times the data is encrypted with symmetric key cryptography and then the symmetric key is encrypted using asymmetric key cryptography.



Digital Signature-Example

- Digital signatures are almost impossible to forge because they are based on number theory; in what is called “public key cryptography”
- Alice uses her private key in the signing algorithm to link a signature to her message and public key



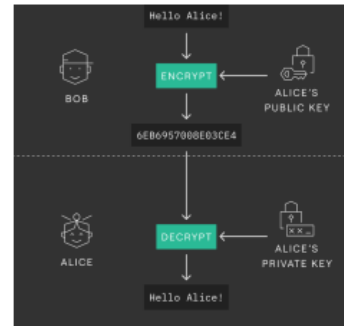
Public key encryption:

<https://www.coinbase.com/cloud/discover/dev-foundations/digital-signatures>

This trick can greatly speed up the asymmetric key cryptography process. To summarize, in this video, we discussed the concept of digital signatures. We discussed some of the key properties of digital signatures. And we also introduced the concept of asymmetric key cryptography which relies on a combination of public and private key where private key remains private with the user and public key is used to verify the digital signatures. In this video, we will carry on with our discussion of digital signature properties and asymmetric key cryptography.

Digital Signature-Example

- Digital signatures are a fundamental building block in blockchains, used mainly to authenticate transactions.
- If Alice wants to send Bob 1 bitcoin, she must sign a transaction spending 1 bitcoin of inputs with her private key and send it to nodes on the network
- The miners, who know her public key, will then check the conditions of the transaction and validate the signature. Once validity is confirmed, the block containing that transaction is ready for finalization by a validator/miner.



Public key encryption:

<https://www.coinbase.com/cloud/discover/dev-foundations/digital-signatures>

We know that a digital signature scheme consists of the following three key important algorithms or methods. The first is generate key method. This generate key method takes a key size and generates a key pair which includes a private key or secret key SK which is private which is used to sign the message. So this secret key or private key will be used to sign the message. Then we have a public key PK.

This public key PK is used for the verification or it is called a verification key which is given to everybody by the initiator, the person who initiates the transaction or the node that initiates the transaction transmits this public key to every member on the node. Now anybody, anyone on the node can use this key to verify the signature. Next, you have sign method. The sign method takes a message and a secret key as input and generates a signature for message for a given message and secret key. This sign method generates a signature.

Lastly, we have a verify method that takes a message, a public key as input and signature and it generates a Boolean true and false values. If it is valid, if the signature is correct and valid, then it generates a Boolean true. If it is not valid, then it generates a Boolean false. Here, we also know that this generate key and sign, they are randomized algorithms because both these generate keys and sign they better be randomized because they have to generate every time they are employed. For example, generate keys method is employed.

They should be generating different keys for different identities every time. So it should be better be randomized. But verify here is more on the deterministic side because it needs to result in a Boolean true and false. It has to match two values and therefore it has to be a

deterministic algorithm. Now let us discuss two very important properties of digital signatures that we require from the digital signature.



Digital Signature Uses

Use of asymmetric-key cryptography in blockchain networks

- Private keys are used to digitally sign transactions.
- Public keys are used to derive addresses.
- Public keys are used to verify signatures generated with private keys.
- Asymmetric-key cryptography provides the ability to verify the ownership of private key

First and foremost, the valid digital signatures must verify. So if I sign a message with my secret key and someone tries to validate that signature using the public key, the signature must verify correctly. And this property is a very basic requirement for signatures to be useful whether it is a digital signature or in real life version of the signature on paper. So this verify method, if somebody uses my public key to verify on the message which I had signed using my secret key on the same message, it should result or it should be able to verify.

So that is one. Second, one should not be able to find the private key using the public key. So this should be infeasible. Lastly, and very importantly, the unforgeability, the signature should be digital signature should be unforgeable. This second requirement that it should be computationally infeasible to forge signatures. That is an adversary who knows my public key and gets to see my signatures on some other message should not be able to forge my signature on some other message for which he has not seen my signature or my signatures are not there on that message.

This unforgeability which is also true in real life, this property generally formalized in terms of a game that we play with an adversary. And this kind of game is very common in cryptographic security proofs to create this property of unforgeability in digital signatures. To summarize in this video, we discussed three very important components of digital signatures. These included the generate key method, generate key, sign method and verify. We also noted that for a digital signature to be good, it has to have two key properties.



Byzantine Generals problem

- The Byzantine general's problem is a game theory problem or a general mathematical puzzle that describes how difficult it is for dispersed parties to reach a consensus in the absence of a central authority.
- How to coordinate when somebody is not acting in good faith or in the best interest of the group!
- There are multiple independent parties (generals in the military metaphor) that must coordinate their actions
- There is no central authority, and therefore, they must coordinate their actions in a decentralized environment
- In a permissionless system, there is no central authority. And if there's no central authority, how does a distributed network, like the distributed set of generals, come to some agreement?

First, the valid signatures must be verified using the public key, they must be able to verify. Anybody who uses public key should be able to verify the message that somebody signed with their secret or private key. Second, the signature should be unforgeable and anybody should not be able to forge using public key should not be able to forge my signatures on a different document and nobody should be able to find my private key using my public key. In this video, we will conclude our discussion about digital signatures and asymmetric key cryptography. Just as written signatures tie a person to a particular document, digital signatures cryptographically link an identity to the message.

So digital signatures are almost impossible to forge because they are based on a number theory, what is called public key cryptography, where users own a public key and a private key, which form a pair. Now public key cryptography uses encryption to guarantee security and protect sensitive key information. The public key here represents the owner's identity and the private key secret to the owner, allowing them to prove that they own the public key. Now let's understand with an example. Suppose Alice wants to send a message to Bob.

She uses her private key in the signing algorithm to link her signature to the message and public key. So she links her private key with the signing algorithm and generates a signature along with the message and the public key. Now nobody can derive Alice private key or forge a valid signature for her using the signature and public key. So her public key is circulated over the network, but nobody can generate or forge her private key and or forge her signature or generate her private key. However, anyone who knows her public key, which is everybody on the network, they can easily verify that the message was signed by her private key.

So everybody on the network can verify through her public key. Please note that digital signatures are a very important fundamental part of blockchain. They are a fundamental building block in blockchains is mainly used to authenticate transactions. When users submit transactions, they must prove to every node or every member node in the system that they are authorized to spend those funds while preventing other users from spending them. And every node in the network will verify the submitted transaction and check all other nodes work to agree on a correct state which we are calling as consensus.

So if Alice wants to send Bob one Bitcoin, she must sign the transaction spending one Bitcoin of inputs with her private key and send her public key to the nodes on the network. Now, the miners who know her public key will then check the conditions of the transaction and validate the signature. So her public keys there on the network with the miners, they will validate the signature. And once the validity is confirmed, the block containing this information about the transaction or any such information is ready for finalization by a validator or miners. So this block once verified will be added to the blockchain.




Byzantine Generals Problem

- Multiple generals besiege Byzantium. They've encircled the city, but they must decide when to assault as a group
- Decentralized systems are susceptible to the Byzantine generals problem, as they lack a dependable source of information
- The Consensus Problem: The consensus problem can be stated in a basic, generic manner: One or more systems may propose some value.
- A reliable computer system must be able to cope with the failure of one or more of its components. A failed component may exhibit a type of behavior that is often overlooked
- In decentralized networks, reliable computer systems must handle malfunctioning components that give conflicting information to different parts of the system.

So to summarize this video, we discussed here the application of asymmetry key cryptography in digital signatures in blockchain networks. First, we noted that private keys are used to digitally sign the transactions. Second, the public keys are used to derive addresses and public keys are used to verify signatures generated by the private keys. And also the asymmetry key cryptography provides the ability to verify the ownership of private key, which means that the user transferring the value to another user is in possession of the private key and capable of signing the transaction. In a series of next few videos, we will discuss the consensus mechanism employed in blockchains.

We start with this video talking about Byzantine Generals Problem. To begin with, the abiding Byzantine Generals Problem is based on a Game Theory analogy. So multiple generals besiege Byzantium and they have encircled the city but they must decide when to assault as a group. They will win if all generals assault simultaneously. However, they will lose if they attack because any letters they transmit or receive could have been intercepted or deceptively sent by Byzantium's defenders and the generals have no secure communication channels with one another. So they may lose in this scenario. Now, how can the generals coordinate the attacks simultaneously? Loyal generals need a safe means to agree on a plan which we call as consensus and carry it out with coordination and the Byzantine Generals Problem must be solved if a dispersed group of nodes, for example computers or other physical devices needs to achieve reliable communications. Now, here decentralized systems are susceptible to Byzantine Generals Problem as they lack a dependable source of information and have no way of confirming the information they get from other network users. In centralized systems an authority is trusted to disseminate accurate information while preventing the spread of erroneous or fraudulent information across the network. For example, in the traditional financial system, banks are trusted to provide clients with accurate balances and transaction histories. If a bank tries to deceive or mislead its consumers, the central bank or government is authorized to take some action and restore the faith.



Consensus Mechanism

- A key aspect of blockchain technology is determining which user publishes the next block.
- In a Blockchain network, multiple nodes across the network verify each transaction and preserve them without having a centralized authority
- In such a situation, why would a user propagate a block that another user is attempting to publish?
- When a user joins a blockchain network, they agree to the initial state of the system

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

16



Consensus Mechanism

- The following properties are then in place
 - The initial state of the system is agreed upon (e.g., the genesis block).
 - Users agree to the consensus model by which blocks are added to the system.
 - Every block is linked to the previous block by including the previous block header's hash digest
 - Users can verify every block independently
- A key feature of blockchain technology is that there is no need to have a trusted third party provide the state of the system
- In the permissioned blockchain networks, there may exist some level of trust between publishing nodes

Now, let's talk about the consensus problem. The consensus problem, suppose you have a collection of computers and want all of them to agree on something. This is what consensus is about. Consensus means agreement. So the consensus problem can be stated in a basic generic manner by stating that one or more systems may propose some value.

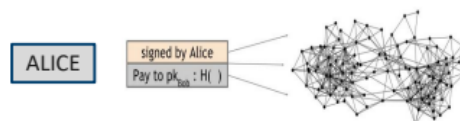
How do we get a collection of computers or collection of nodes or members or devices to agree on exactly one of those proposed values? A reliable computer system must be able to cope up with the failure of one or more of its component. A failed component may exhibit a type of behavior that is often overlooked, particularly namely sending conflicting information to different parts of the system for example. The problem of coping with this type of failure is expressed abstractly as the Byzantine Generals Problem. Now, reliable computer systems must handle malfunctioning components in decentralized networks. The reliable computer systems must handle malfunctioning components that give conflicting information to different parts of the system.

This situation can be expressed abstractly in terms of group of generals of the Byzantine army camped with their troops around the enemy city and communicating only by messenger and the generals must agree upon a common battle plan. However, one or more of them may be traitors who will try to confuse the others. The problem is to find an algorithm or a system of coordination to ensure that loyal generals will reach an agreement. So let us summarize this Byzantine Generals Problem. The Byzantine Generals Problem is a game theory problem or a general mathematical puzzle of sort that describes how difficult it is for dispersed parties to reach a consensus in the absence of a central authority.



Consensus Mechanism

- When Alice wants to pay Bob, what she actually does is broadcast a transaction to all of the Bitcoin nodes that comprise the peer-to-peer network.
- What exactly is it that the nodes might want to reach a consensus on in the Bitcoin network?
- How exactly do nodes come to a consensus on a block?
- In Bitcoin, we do consensus on a block-by-block basis



Alice broadcasts the transaction to the entire peer-to-peer network.

Source: Bitcoin and Cryptocurrency Technologies (Page 53)
<https://press.princeton.edu/books/hardcover/9780691171692/bitcoin-and-cryptocurrency-technologies>

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

18

So how can members of a network agree on a specific reality when no one can verify the identities of other members? So the question arise, how to coordinate when somebody is not acting in good faith or in the best interest of the group. There are multiple independent parties, like the generals in the military metaphor that must coordinate their actions. There is no central authority and therefore they must coordinate their actions in a decentralized environment. Lastly, in a permissionless system, there is no central authority and if there is no central authority, how does a distributed network like the distributed set of generals come to some agreement and this requires some kind of consensus mechanism that will be employed to solve these kind of Byzantine Generals Problem and we will see and discuss such consensus mechanisms in next set of videos. In this video, we will introduce the consensus mechanism.

In blockchain technology, a key aspect is determining which user publishes the next block. This is solved through implementing one of many possible consensus models. For permissionless blockchain networks, there are generally many publishing nodes competing at the same time to publish the next block. They usually do this to win cryptocurrency and transaction fees. In native currency, they are generally mutually distrusting users that may only know each other by their public addresses.

Each publishing node is likely motivated by a desire for financial gain, not the well-being of the other publishing nodes or even the network itself. So in a blockchain network, multiple nodes across the network verify each transaction and preserve them without having a centralized authority. This verification and insertion of transactions are achieved through distributed cryptographic mechanism called consensus protocol. Now given the situation we described, why would a user propagate a block that another user is attempting

to publish? Also, who resolves conflicts when multiple nodes publish a block at approximately the same time? To make this kind of setup work, blockchain technologies use consensus models to enable a group of mutually distrusting users to work together. Here, when a user joins a blockchain network, they agree to some initial state of system.



Proof-of-Work (PoW) Consensus Model

- If you deposit a cheque in your savings account, how do you know that you'll be credited for the accurate amount?
- Cryptocurrencies do not have centralized gatekeepers to verify the accuracy of new transactions and data that are added to the blockchain
- Proof of work is a technique used by cryptocurrencies to verify the accuracy of new transactions

This is recorded in the only pre-configured block, we call it as the Genesis block. Every blockchain network has a published genesis block and every block must be added to the blockchain after it. So this is the genesis block or the first block and every block comes after that only. Based on some pre-agreed consensus model, so based on that consensus model, these blocks will be added and regardless of the model, however, each block must be valid and thus can be validated independently by each blockchain network user or the nodes on the network. By combining the initial state and the ability to verify every block since then after this genesis block, users can independently agree on the current state of blockchain and please note that if there were ever two valid chains presented to a full node, there are two claims, claim 1 and claim 2, c1 c2.

The default mechanism in most blockchain network is that the longer chain, the chain which is lengthier, longer chain, the lengthier chain is viewed as the correct one and will be adopted. This is because it has the most amount of work put into it. So verification and all the effort that has gone through consensus mechanism has gone into a lengthier chain mode. So this is given the first preference. So based on this, the following properties become part of the blockchain system.

First, the initial state of the system is agreed upon or what we call as Genesis block. So the conditions or rules are set there. Next, the users agree to the consensus model by which

the blocks are added to the system. Every block is linked to the previous block. Headers hash digest which we have, this headers hash digest we have already discussed.


This is except for the first Genesis block because it has no previous blocks and for which the hash of the previous block header is usually set to zeros because there is no block previously. Now users can verify each block independently. In practice, the software handles everything and the users do not need to be aware of all these details. Now a key feature of blockchain technology is that there is no need to have a trusted third party or central party to provide the state of the system because every user within the system can verify the system's integrity. To add a new block to the blockchain, all nodes must come to a common agreement over time.

However, some temporary disagreement is permitted. For permissionless blockchain networks, the consensus model must work even in the presence of possible malicious users since these users might attempt to disrupt or take over the blockchain. Please note that for the permissioned blockchain networks, legal remedies may also be used if there is a malicious user because there is a central counterparty, that central party, for example, there is a central party which will make a check or put penalty on malicious users in case of permissioned blockchain networks. So in some permissioned blockchain networks, there may exist some level of trust between publishing nodes. In this case, there may not be the need for a resource intensive, for example, high computation time investment and so on, etc. for consensus model to determine which participant adds the next block to the chain.

Generally, as the level of trust increases, the need for resource usage as a measure of generating trust decreases. Now for some permissioned blockchain implementations, the view of consensus extends beyond ensuring validity and authenticity of the blocks, but it encompasses the entire systems of checks and validations from the proposal of a transaction up until its final inclusion in the block. Let us understand this with the help of simple example and what does it mean in the context of a currency like Bitcoin? To understand how distributed consensus could work in Bitcoin, please recall that Bitcoin is a peer-to-peer system. So for example, if Alice wants to pay Bob, what she does is broadcast the transaction to all of the Bitcoin nodes that comprise the peer-to-peer network. Now incidentally, you have noticed that here the Alice here broadcast to all the nodes here.


It is signed by her private key and the message also contains to all the nodes they get her public key and you may have noticed that this broadcast to the transaction to all the peer-to-peer nodes, but Bob's computer is nowhere in this. Bob is not there. It is of course possible that Bob is running one of the nodes in the peer-to-peer network. Maybe he is there and in fact, if he wants to be notified that this transaction did in fact happen and that

he got paid, running a node may be a good idea. Notwithstanding this discussion, there is no such particular requirement that Bob be on this node or listening to the network.



Proof-of-Work(PoW)

- **New transactions are grouped together.** Users buy and sell cryptocurrency, and the data from these transactions are pooled into a block.
- **Miners compete to process the new block.** Crypto miners compete to be the first to solve a complex math problem. By showing proof that they've undertaken the computational work—referred to as a hash—earns the miner the right to process the block of transactions.
- **One miner is chosen to add the new block.** There is a degree of randomness in deciding which miner wins the right to process the block. The winner is awarded new cryptocurrency coins and adds a new block to the blockchain.
- **Cryptocurrencies that Use Proof of Work:** Bitcoin, Dogecoin, Bitcoin Cash, Litecoin, Monero.



The diagram illustrates the Proof-of-Work process. It features a central circle labeled 'Proof-of-Work' with four square nodes. Clockwise from the top, the nodes are: 'Miners' (with a miner icon), 'Block' (with a blue cube icon), 'Block Puzzle' (with a puzzle piece icon), and 'Proof-of-work' (with a dollar sign icon). A red arrow points from 'Block' to 'Block Puzzle', and another from 'Block Puzzle' to 'Proof-of-work'. A third red arrow points from 'Proof-of-work' to 'New Block Broadcasting' (with a miner icon), which then points to 'Miner Verification' (with a padlock icon), which points to 'Process Start over' (with a gear icon), which points to 'Transaction' (with a Euro symbol icon), which points back to 'Miners'. Below the diagram, the text reads: 'Working of proof-of-work' and 'Source: A Novel Optimization for GPU Mining Using Overclocking and Undervolting https://www.mdpi.com/2071-1050/14/14/8708'.

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

Running a node is not necessary for Bob to receive the fund in this transaction. The Bitcoins will be his whether he is operating a node on the network or not. Now what exactly is that the node might want to reach a consensus on in that network? Given that a variety of users are broadcasting these transactions like Alice on the network, the nodes must agree on exactly which transactions were broadcast and the order in which these transactions happened. This will result in a single global ledger for the system. Second, how exactly do these nodes come to consensus on a block? One way to do this at regular intervals, say every 5 or 10 minutes, every node in the system proposes its own outstanding transaction pool to the next block and the nodes execute some consensus protocol where each node input its own proposed block.

Now some nodes may be malicious and put invalid transactions in their blocks, but we might assume that other nodes will be honest and if the consensus protocol succeeds, a valid block will be selected as the output. So a valid block will be added to the chain. New block will be added. Even if the selected block was proposed by only one node, it is a valid output as long as the block is valid. And now there may be some valid outstanding transaction that did not get included in this block.

Probably some transaction may need to wait like we wait for the bus at bus stand. But this is not a problem. If some transaction somehow didn't make it into this particular block, it could just wait and get into the next block. When the next block comes, the transaction can get added to that.

Lastly, in Bitcoin, we do this consensus on a block by block basis. So at any given point, all the nodes in a peer to peer network, these nodes have a ledger consisting of a sequence of blocks, each containing a list of transactions that we have reached consensus on. Additionally, each node has a pool of outstanding transactions that it has heard about, but have not been included on the blockchain. For these transactions, consensus had not yet happened. And so by definition, each node might have a slightly different version of the outstanding transaction.



How Does PoW Work?

- Proof of work requires miners to guess a random number that should give them the right hash for that block of transactions. This process involves to key inputs.: **nonce** and **hash** function
- Nonce – it's a random number used only once. In the case of Bitcoin, that number is an integer.
- Hash – it's an algorithm or very complicated formula that converts any sequence of characters into a string of 64 chars or numbers.
- Every block in the blockchain has its hash (id). To verify the next block, you will take that hash and add the current block of transaction
- The next step would be to take a nonce and add it to the end of that block of text

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

24

In practice, this occurs because the peer to peer network is not perfect. So some nodes may have heard about a transaction that other nodes have not heard about. To summarize, in this video, we discuss the consensus mechanism through which a new transaction gets added to the block and becomes part of the blockchain. In this video, we will continue with our discussion about consensus mechanism. We will discuss a very important component of this that is proof of work.

Think of a conventional bank account. If you deposit a check in your savings account, how do you know that it will be credited for the accurate amount? How does the writer of the check press that they will be only debited for the amount that they wrote on the check? The value of a bank is that all the parties to a transaction trust the bank to accurately move money around. With cryptocurrencies, there are no such bankers or financial institutions to ensure trust. Instead, miners and proof of work mechanism guarantee transparent and accurate transactions. For blockchains that use proof of work, miners are the guardians and facilitators that make the system run smoothly and accurately. Now cryptocurrencies do not have centralized gatekeepers to verify the accuracy of new transactions and data that are added to blockchain.

Instead, they rely on distributed network of participants to validate incoming transactions and add them as new blocks on the chain. Proof of work is a consensus mechanism to choose which of these network participants, called miners, are allowed to handle this lucrative task of verifying new data. It is lucrative because the miners are rewarded with the new crypto when they accurately validate the new data and don't cheat the system. Please note that the proof of work here is a function that is hard to compute but easy to check.

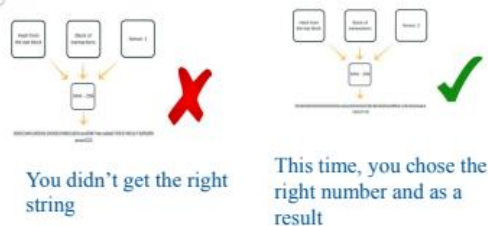


How does PoW work?

- When you have all that, you are ready to start calculations. For calculations, you use the hash function and change the random number until you get a string that has a certain number of zeroes in front of it.

$\text{SHA256}(\text{"blockchain"} + \text{Nonce}) = \text{Hash Digest starting with "000000"}$

- Everything is ready and you send everything to the hash function
- A machine or computer must keep incrementing a nonce until it finds the right one
- Let's say someone wants to check if Node A did the required work. He will simply use the block string that Node A got after validation and take its nonce number



INDIAN INSTITUTE OF TECHNOLOGY KANPUR

25

So it is hard to compute but easy to check. The function has a message, a recipient address and a few other parameters. This proof of work protocol became popular when Satoshi Nakamoto released Bitcoin whitepaper and the proof of work was the most significant idea behind it. With that protocol, Satoshi introduced the idea of how it can be used to allow trustless and distributed consensus. Proof of work is a technique used by cryptocurrencies to verify the accuracy of new transactions that are added to a blockchain. The decentralized networks used by cryptocurrencies lack any central governing authority.

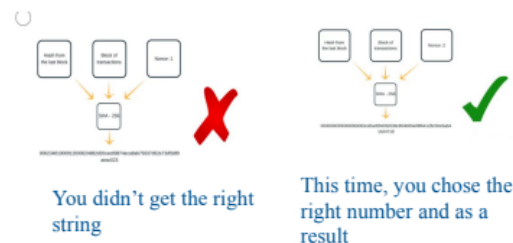
So they employ a proof of work mechanism to ensure the integrity of new data. A proof of work is a sort of consensus algorithm in which it is costly and time consuming to produce a piece of data but it is easy for others to verify that data is correct. And the most popular cryptocurrency Bitcoin uses a hash-cash proof of work system. Although the initial hash-cash idea was to fight against the email spammers, Satoshi applied this idea to Bitcoin transaction verification through proof of work mechanism as well. Let us discuss this proof of work and mining. For a block to be accepted by the network, miners have to complete a proof of work to verify all the transactions in the block.

The difficulty of this work is not always the same. It keeps adjusting so that new blocks can be generated every 10 minutes maybe. There is a very low probability of successful generation. So it is unpredictable which worker in the network will produce the next block. So there is a randomness in this. In a network, users send each other coins and information and the ledger gathers the transactions into blocks.



How does PoW work?


- Proof of work ensures that blocks can't be added to the blockchain without executing the necessary work
- Once a publishing node has performed this work, they send their block with a valid nonce to full nodes in the blockchain network
- To execute PoW we have to spend high amounts of electricity and computer power
- Also, people with large computing power have unfair advantage in the system



But someone should take care of all the transactions and validate them. In every blockchain, some nodes are doing the validation. The example of blockchain the miners are nodes. Now the work in proof of work is the key. The system requires miners to compete with each other and race it to become first to solve that some kind of arbitrary mathematical puzzle to prevent anybody from gaming the system and the winner of this race is selected to add the newest batch of the data or transactions in the form of block to the blockchain. Now winning miners only receive their reward of new cryptocurrency after other participants in the network verify that the data being added to the chain is correct and valid.

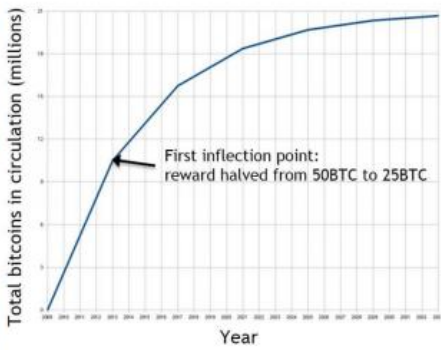
So in the proof of work model, in this kind of model, a user publishes the next block by being first to solve this computationally intensive puzzle. A user publishes the next block by being the first to solve a computationally intensive puzzle and the solution to this puzzle is the proof that they have performed the work. And the puzzle is designed such that solving the puzzle is difficult but checking that a solution is valid is easy. And this mechanism enables all other full nodes to easily validate any proposed next blocks and any proposed block that did not satisfy the puzzle would be rejected. Lastly, one of the issues that had prevented the development of an effective digital currency in the past was called the double spend problem because cryptocurrency is just data so there needs to be a mechanism to

prevent users from spending the same units in different places before the system can record the transactions.



Incentives and Proof of Work

- 1. Block Reward:**
 - According to the rules of Bitcoin, the node that creates a block gets to include a special transaction in that block.
 - After every 210,000 blocks (or approximately four years), the block reward is cut in half. It means that there is a finite total of 21 million bitcoins maximum (by 2140)
- 2. Transaction fees:**
 - The transaction fee is purely voluntary, but as the block reward starts to run out, it will become more and more important, almost mandatory, for users to include transaction fees in order to get a reasonable quality of service



Source: Bitcoin and Cryptocurrency Technologies (P.63)
<https://press.princeton.edu/books/hardcover/9780691171692/bitcoin-and-cryptocurrency-technologies>

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

While you would have a hard time spending the same dollar bill on two separate purchases, anyone who's duplicated a computer file by copying and pasting and probably imagine how you could spend digital money twice, even 10 times or more. So Satoshi's consensus mechanism solved the double spend problem by incentivizing miners to verify the integrity of new crypto transactions before adding them to the distributed ledger that is blockchain. And here proof of work helps preventing the double spending problem. Let us understand a proof of work mechanism.

It requires miners to use computing resources for the privilege and here how it works. We will state with the help of example and diagram. First and foremost, new transactions are grouped together so users buy and sell cryptocurrency so the transactions take place and the data from these transactions are pulled into a block. So this block would contain a set of data and there would be miners waiting. Now these miners compete to process the new block.

So the crypto miners compete to be the first to solve this some kind of mathematical problem. So there's a block puzzle and these miners they compete to solve this mathematical problem. Once they solve, they show the proof of work that they have undertaken and completed this computational work by solving this puzzle. So they have a proof of work which is referred to as hash and they use this proof of work to show other nodes in the blockchain and earn the right to process the block of transactions. So the whosoever, one of the miners who process and provide the proof of work first gets the right to process the block of transactions.

One miner is chosen to add this new block. So there is a degree of randomness in deciding which miner wins the right to process the block and whosoever is the winner is awarded the new cryptocurrency. So he gets award in the form of native cryptocurrency and that's a new block to the blockchain. So the new block is added by this miner. Cryptocurrencies that use proof of work include Bitcoin, Dongycoin, Bitcoin Cash, Litecoin, Monero among others.

To summarize this video, miners work to solve complex math problems to earn a reward. These miners, these are laborious problems that require significant computing power and energy to solve. Since miners have invested considerable resources and money in the computer equipment and energy cost required, they are motivated to accurately validate these transactions. And in a proof of work kind of blockchain, the participants are ready to spend computational power to solve cryptographic problems and get the right to add new blocks to the blockchain, the winning miner, the miner who has won this race. And the way these nodes authorize transactions or the miners authorize these transactions depend on the consensus algorithm.

It need not have proof of work. Later we will discuss something like proof of stake mechanism also. But for example, in Bitcoin, it is proof of work. And to summarize this proof of work mechanism, it is a system that ensures security and consensus throughout the blockchain network. It is evident that participants who validate block have invested significant computing power in doing so. In this video, we will discuss the proof of work or POW consensus mechanism. To begin with, proof of work requires miners to guess random number that should give them the right hash for that particular block of transactions.

This process has two key inputs. One is nonce and hash function. Nonce is sort of random number used only once. So random number used only once or number used only once. That can be combined with the data to produce different hash digests per nonce. So it's a random number used only once. It's a sort of cryptographic number or cryptographic nonce that can be combined with data to produce different hash digests per nonce.

For example, if you apply hash to some data which is there in a block plus this nonce, you will get a digest or a hash code kind of as output. This is hash function. Now only changing the nonce value, only changing this nonce value provides a mechanism for obtaining different digest values or hash codes. So every time you change this nonce, you will get a different digest or hash code while keeping the same data. So with the same data you can generate different hashes and this technique is integral and employed heavily in the proof of work consensus model.

Next we have hash function. Hash is an algorithm. It's a very complicated formula that converts any sequence of characters into a string of 64 characters or numbers. A specific cryptographic hash function used in many blockchain implementations like Bitcoin is Secure Hash Algorithm or SHA which is a secure hash algorithm SHA with an output size of 256 bits. So it is also called SHA-256. Many computers support this algorithm in hardware making it faster to compute and SHA-256 has an output of 32 bytes which means 1 byte equal to 8 bits so 32 bytes equal to 256 bits. So generally it is displayed as a 64 character hexadecimal string in a 64 character hexadecimal string format and every block in the blockchain has its own hash ID that is a string that someone got when they verified that block.

So when you want to verify the next block you will take that hash add the current block of transactions and you will get a big block of text. So you take that hash function you add the current block of transaction and you will get a big block of text. Now the next step would be to add a nonce to it so the third component is nonce and add it to the end of the block of text. Now you have a big block of text which contains the hash of the previous block so you have the hash of previous block the block of text plus random number nonce so you have 3 key inputs.

Now when you have all these inputs ready you are ready to start the calculations. For calculations you use the hash function keep changing the random number until you get the string that has a certain number of zeros in front of it. So however this requires 10 to the power 21 calculations given the architecture of SHA secured hash algorithm SHA-256 so you need to have 10 to the power 21 calculations to find the right number that's not a small number it takes around 10 minutes to find the right number that will give you the needed string the code the hash code. For a hash function Bitcoin uses SHA-256 algorithm now let's suppose now that you have everything to start computing and the hash from the previous block has 18 zeros so hash from the previous block contains 18 zeros and you have a bunch of hundred transactions in the block you pick a random number maybe one or two so you pick a random number and now you have everything ready you send everything to hash function so you have the block of transactions you have the random number you send it to the hash function and you get the hash digest with a certain number of zeros in it. Now in this particular illustration you can see that you did not get the right string so your string the number of zeros in your string does not match the resulting string maybe starts with only three zeros while desirable string that you want starts from 18 zeros so that means your random number nonce was not correct so we'll start the calculations again maybe pick another number 2 first we pick 1 then we pick 2 now this time you chose the right number let's say and as a result you got the string that started with 18 zeros so you got the string with 18 zeros so your hash function when applied to the hash from the previous block so this is your hash from the previous block this is your hash from the previous block previous block this is the block of transaction latest block that we are trying

to verify it's block of transactions and this is the news and when it is added to our hash function as HA256 we get the code hash digest with 18 zeros so we are able to verify it. Now that you have verified this block and you are the first one to verify it you'll get it as a reward and this block goes to you this new block that is verified goes to you for processing so sort of you sort of won it.

Now a machine or computer must keep incrementing or changing their nonce random number until you find the right code it means that a computer has to sort of brute force that number and generate millions of hashes per second to generate one particular hash code or digest that will have the same number of zeros in it and it's a costly and time consuming process to do this kind of brute force mechanism which we are calling as proof of work for a block this kind of consensus mechanism but while generating this digest and random number and nonce is easy is difficult and time consuming it is easier for someone to verify so if you give the nonce to somebody with all the inputs it is easier for them to verify this code so generating the nonce is difficult but easier to verify the output. Now let's say someone wants to check if node A did the right work did the right proof of work all they need is simply use the block string that node A got after validation take its nonce number then they will apply the hash function and if the result has the correct number of leading zeros that 18 zeros everything is fine so they have verified your proof of work. In this fashion proof of work ensures that the blocks can be added to the blockchain without executing the necessary proof of work or consensus that way a malicious or evil node can't easily validate blocks and add whatever they want into the blockchain so if they try to other parts in the network would just dismiss that block because it would not be verified it would not match and everyone would know that this block is not valid. Now once a publishing node has performed this work they send their block with a valid nonce to full nodes in the blockchain for verification the recipient full nodes verify that the new block fulfills the puzzle requirement and then add the block to their copy of the blockchain and resend the block to their peer nodes also so in this manner this new block gets quickly distributed throughout the network of participant nodes and verification of node is easier since only a single hash needs to be done to check if the node or this information the number nonce and other inputs solve the puzzle. Lastly proof of work is a great consensus algorithm but is not perfect to execute proof of work you have to spend high amounts of electricity computation power just to brute force the hashes so the power used for proof of work is just a waste of resources for example just imagine if some cryptocurrency that uses this POW proof of work consensus mechanism goes for mainstream adoption maybe like a CBDC how much electricity would be used to have the network running.

Another problem here is the centralization and mining pools someone who mines with just one CPU will not have a good chance for a reward but someone who has a mining pool with thousand CPUs will have a better chance so there is a threat that a few people with large computing power have unfair advantage in the system which means if they have large

computing power resources they can meet the same at the same desk they can small number of people they can go to same desk agree together with 50% or more volume of computing power and change the blockchain according to their wishes. To summarize in this video we discussed the proof of work consensus mechanism how it functions we also discussed some of the pros and cons of this mechanism. In this video we will discuss two key incentives in the proof of work consensus mechanism that is block reward and transaction fees. Please note that we can reward each of the nodes that created the blocks that did end up on the long term consensus chain that is incentivize nodes to behave honestly by paying them in units of this currency the native currency.



Proof-of-Stake Consensus Model

- Staking is when people agree to lock up an amount of cryptocurrency in exchange for the chance to validate new blocks of data to be added to a blockchain
- The simple way to look at staking is like interest income that requires you to complete a task to earn the interest
- The proof of stake (PoS) model is based on the idea that the more stakes user has invested into the system, the more likely they will want the system to succeed
- With this consensus model, there is no need to perform resource intensive computations (involving time, electricity, and processing power) as found in proof of work

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

Now there are two separate incentive mechanisms in Bitcoin chain for example we will discuss one by one. The first method is block reward method according to the rules of Bitcoin the node that creates the block gets to include a special transaction in that block this transaction is a coin creation transaction and the node can also choose the recipient address of this transaction of course that node will typically choose an address belonging to itself because they have mined transaction successfully so you can think of this as a payment to the node in exchange for the service of creating a block on the consensus chain. Now this is a very interesting thing let's focus on it you may be wondering why the block reward incentivizes honest behavior it may appear that this node gets the block reward regardless of whether it proposes a valid block or behaves maliciously but this is not true think about it how will this node collect its reward that will only happen if the block in question ends up on the long term consensus branch because just like every other transaction the coin creation transaction will only be accepted by other nodes if it ends up on the consensus chain and that's the key idea behind this incentive mechanism it is a



Summary and Concluding Remarks

- One such mechanism is proof of work model where miners are incentivized to act in good faith and verify the transaction by solving a mathematical puzzle
- The incentive is in the form of native currency through transaction fee and block rewards. However, this approach is energy intensive and requires considerable amount of electricity consumption
- The other approach is proof of stake approach which gives preferential treatment to those miners who have greater stakes in that blockchain in the form of their ownership of the native currency

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

very subtle but powerful trick it incentivizes nodes to behave in whatever way they believe they will get other nodes to extend their blocks so if most of the network is solving the longest valid branch rule it incentivizes all nodes to continue to follow that rule and that's the bitcoins first incentive mechanism now there is a catch after every two lakh ten thousand blocks or approximately four years the block reward is cut in half it means that there is a finite total of 21 million bitcoins that would be created by 2140 year and that is the max it is important to note that this is the only way in which new bitcoins are allowed to be created and there is no other coin generation mechanism and that's why we are saying that only 21 million is the final total number as the rule stands now for how many bitcoins there can possibly maximum the maximum number so this new block creation reward is actually going to run out in 2140 2140 it will run out as thing stands now so no new cryptocurrency will be created so does it mean that the system will stop working in 2140 and become insecure because nodes are no longer they no longer have been sent up to behave honestly not quite the block reward is only the first of the two mechanism that there is transaction fee now here we can see this the geometric kind of movement and the decline in block Bitcoin creation and we can see it is going to end in year 2140 the total supply is going to step stabilize here because no new bitcoins will be created so let's discuss the transaction fee incentive the second incentive mechanism is called transaction fee mechanism the creator of any transaction can choose to make the total value of the transaction total value of transaction output less the total number value of its inputs so who so creates the block that first puts the transaction into the block chain gets to collect the difference which acts as a transaction fee so if you are a node that is creating a block that contains let's say 200 transactions then all the values of the sum of these 200 transaction fees all the sum of those 200 transaction fees is paid to the address that you put into that block the transaction fee is purely voluntary but you would expect based on our

understanding of the system that as the block reward starts to run out and no new cryptocurrency creation will happen it will become more and more important almost mandatory for users to include a transaction fee in order to get a reasonable quality of surface so they it will become almost mandatory as the cryptocurrency production stops and it reaches its maximum supply and to a certain degree this is already starting to happen now but is yet unclear precisely how the system will log because it really depends on lot of game theoretical manner which hasn't been fully worked out yet so this is an another interesting open area of research so to summarize this video we discussed two ways where proof of work mechanism incentivizes miners first is the block reward but we noted that this there is a limitation to this and as the supply of bitcoins maximizes or any such cryptocurrency which where there is a maximum supply is limited to a certain number this block reward will be restricted and afterwards we have another mechanism which is transaction fees so miners get paid this transaction fee in the form of native currency by the by for providing the service of mining by those who are originating the transaction or want to transact they have to pay a small fraction as a transaction fees and it is expected that as the production of that native currency stops no new coins are or currency is created then this transaction fee will become as mandatory and very important component of incentive for such proof of work mechanism in this video we'll discuss an alternative to proof of work consensus model which is proof of stake consensus model staking is when people agree to lock up an amount of money or cryptocurrency in exchange for chance to validate in this case new blocks of data to be added to a blockchain these validators or stakers put their crypto money into a smart contract that is held on the blockchain the blockchain algorithm selects validators to check each new block of data based on how much crypto they have staked the more you stick the better your chance of being chosen as minor to do the work when the data that's being cleared by the validator is added to the blockchain they get newly minted or newly created crypto money as a reward now a simple way to look at this staking is like interest income the interest that you are earning that requires you to complete a task to earn more interest by checking blockchain transaction so if I validate only good transactions I earn interest on my assets if I include bad transactions then I'll be penalized and I lose some of my assets so if a validator submits bad data or fraudulent transactions that they could be punished by slashing or it means that the stake is burned or removed meaning it is sent to an unusable value address on blockchain where nobody has access and renting this money useless forever so this proof of stake works because validators are saying that they have so much faith in the legitimacy of the transaction that I am billing to back it up with my own crypto money or the native currency money and verified transactions earn a cryptocurrency reward in proportion to the size of the stake thus this proof of stake model is based on the idea that the more stakes user has invested into the system the more likely they will want the system to succeed and the less likely they will want to subvert it so the stake is often an amount of cryptocurrency that the blockchain network user has invested into the system maybe through various means

such as by locking it via special transaction type or by sending it to special address or holding it within the special value software once a stake the cryptocurrency is generally no longer able to be spent and proof of stake blockchain networks use that amount of stake a user has as a determining factor for publishing new blocks thus the likelihood of a blockchain network user publishing a new block is tied to the ratio of this stake to the overall blockchain network amount of stake cryptocurrency so the more stake you have the more chances you get or more likelihood to verify a transaction or take part in the verification and later which will earn you more rewards so to summarize this proof of stake mechanism with this kind of consensus model there is no need to perform resource intensive computations which involve time electricity and processing power as you find in proof of work because this consensus model utilizes fewer resources some blockchain networks have decided to forego a block creation reward now so these systems are designed so that all the cryptocurrencies already distributed among users rather than new cryptocurrency being generated at a constant pace and in such systems the reward for block publication and block verification is usually the earnings from transaction fees transaction fees to summarize this lesson digital signatures ensure integrity and validity of the data that is part of blockchain transaction this is ensured through asymmetric key cryptography involving a secret private key and public key the two key attributes of digital signatures are first the public key must verify the transaction data and using public key it should be infeasible to find the private key also the signature should be unforgeable the three important applications of asymmetric key cryptography are first address derivation second digitally sign a document and third verification of signatures and thus the ownership a very important problem addressed by blockchain design is that of byzantine general's problem that is how to coordinate verify and secure the actions of a group where members are located at different locations and there's a threat of malicious attack which can change or modify some important communication or strategic plan of the group this requires development of a consensus mechanism across members and members that does not have a central counterparty and are self-capable of indulgently providing a mechanism to ensure and secure and verify the transaction or such information one such mechanism is proof of work model where miners are incentivized to act in good faith and verify the transaction by solving a mathematical puzzle the incentive is in the form of native currency through transaction fee and block rewards however this approach is energy intensive and requires considerable amount of electricity consumption the other approach is proof of stake approach which gives preferential treatment to those miners who have greater stakes in that blockchain in the form of their ownership of the native currency .